

JAVA 练习题

一、单选题

- 1、下列关于 java 语言的叙述中，正确的是 _____。
A. 机器语言 B. 汇编语言 C. 面向过程的语言 **D. 面向对象的语言**
- 2、下列特点中不属于 Java 的是 _____。
A. 多线程 **B. 多继承** C. 跨平台 D. 动态性
- 3、编译 HelloWorld.java 的正确命令是_____。
A. java HelloWorld.class B. java HelloWorld.java
C. javac HelloWorld.java D. java HelloWorld.class
- 4、运行 HelloWorld.java 的正确命令是_____。
A. java HelloWorld B. javac HelloWorld.java
C. javac HelloWorld.class D. javac HelloWorld.class
- 5、下列命令中，哪个命令是 Java 的编译命令？ _____。
A. javac B. java
C. javadoc D. appletviewer
- 6、下列说法哪个正确？ _____。
A. 一个程序可以包含多个源文件
B. 一个源文件中只能有一个类
C. 一个源文件中可以有多个公共类
D. 一个源文件只能供一个程序使用
- 7、Java 程序默认引用的包是_____。
A. java.ext 包 B. java.awt 包 **C. java.lang 包** D. java.util 包
- 8、下面哪种类型的文件可以在 Java 虚拟机中运行 _____。
A. .java B. .jre C. .exe **D. .class**
- 9、下列关于 Java 特点的描述中，错误的是_____。
A. Java 是一门面向对象的语言
B. Java 具有自动垃圾回收的机制
C. Java 可以运行在 Window 和 Linux 等不同平台上
D. Java 中只支持单线程运行
- 10、关于方法 main（）的说法哪个正确？ _____。
A. 方法 main（）只能放在公共类中
B. main()的头定义可以根据情况任意更改
C. 一个类中可以没有 main()方法
D. 所有对象的创建都必须放在 main()方法中
- 11、java 中 main()方法的返回值是_____。
A. string B. int C. char **D. void**
- 12、以下关于 Java 程序的叙述中正确的是_____。
A. 在 Java 程序中，main 函数必须位于程序的最前面
B. Java 程序的每行只能有一条语句
C. 在对一个 Java 程序进行编译的过程中，可发现注释中的描写错误
D. Java 程序中必须有一个主函数

13、Java application 中的主类需包含 main 方法，以下哪项是 main 方法的正确形参？_____。

- A. String args
C. Char arg
- B. String ar[]**
D. StringBuffer args[]

14、 void 的含义是_____。

- A. 方法没有返回值
B. 方法体为空
C. 没有意义
D. 定义方法时必须使用

15、 return 语句_____。

- A. 只能让方法返回数值
B. 方法都必须含有
C. 方法中可以有多句 **return**
D. 不能用来返回对象

16、如果一个 Java 文件内定义了多个类，关于该文件的命名，以下说法的正确的是_____。

- A. 必须以调用或继承层级最高的类命名
- B. 如果有多个 `public` 修饰的类，则这些类名可以做文件名
- C. 可使用第一个或者最后一个类的类名命名
- D. 如果没有 `public` 类，可以随意命名

17、如果一个 java 文件中定义了 public class A 和 class B, 但都不包含 main()方法, 以下说法正确的是_____。

- A. 该 java 文件如果可以独立编译，就可以独立运行
- B. 该 java 文件即使可以独立编译，无法独立运行
- C. 需要在 A 和 B 类中都添加 main()方法，该文件编译后才可能独立运行
- D. 只有在 A 类中添加 main()方法，该文件编译后才可能独立运行

18、下列各种 java 中的函数的定义，哪种在 java 语言中是正确的_____。

- A. void myFun(int X=1){ }
- B. void myFun(int & X){ }
- C. void myFun(int X){}
- D. void myFun(int * X){ }

19、编译 Java Application 源程序文件将产生相应的字节码文件，这些字节码文件的扩展名为 _____。

- A. .java B. .class C. .html D. .exe

29、设 $x=1, y=2, z=3$, 则表达式 $y+=z- - / ++x$ 的值是_____。

- A. 3 B. 3.5 C. 4 D. 5

21、设 $a=8$, 则表达式 $a>>2$ 的值是_____。

- A. 1 B. 2 C. 3 D. 4

22、若有 `int a=1, b=2, c=3, d=4; boolean m, n;`

执行表达式(m=a>b)&&(n=c++>d)后 c 的值为_____。

- A. 1 B. 2 **C. 3** D. 4

23、若有 `double x=2,y`;则表达式 `y=x+5/2` 的值是_____。

- A. 3 B.4 C.4.5 **D.4.0**

24、int a=12,b; b=a>>2;则 b、a 的值为_____。

- A. 3、3 **B.3、12** C. 6、12 D. 24、3

25、表达式(short)8/9.2*5 的值的类型为_____。

- A. short B. int C. double D. float

26、若 $x = 5, y = 8$, 则表达式 $x|y$ 的值为_____。

- A. 3 B. 13 C. 0 D. 5

27、若有定义：short a=10, b=20, c; int d; float f; 下列运算正确的是_____。

- A. f=a*3.14; B. c=a+b; C. c='\A'+32; D. c='A'+a;

28、以下哪个不是 Java 的原始数据类型_____。

- A. int B. Boolean C. float D. char

29、设 int X=1,则 “X == 1” 表达式之值为_____。

- A. 1 B. false C. true D. null

30、整型数据类型中，需要内存空间最少的是_____。

- A. short B. long C. int D. Byte

31、指出正确的表达式_____。

- A. byte=128; B. long l=0xffffL; C. Boolean=null; D. double=0.9239d;

32、下列初始化正确的是_____。

- A. int a = 2021.12f; B. float b = 2021.12f; C. char c = 2021.12f; D. long d = 2021.12f;

33、下列关于 Java 数据类型转换的说法中，错误的一项是_____。

- A. 表达式 String.format("%.4f", Math.PI)的值为 “3.1415”
B. boolean 类型无法被类型转换
C. int 类型转换为 byte 类型可能会发生溢出
D. double 类型转换为 float 类型必须使用强制类型转换

34、Java 中定义常量的保留字是_____。

- A. const B. final C. finally D. Native

35、下列属于非法标识符的是_____。

- A. _xyz B. 2xyz C. _2xy D. __

36、某个类中定义了成员变量

short s1, s2;

long l1;

float f1, f2;

这些成员变量占用的内存空间是多少? _____。

- A. 20 B. 18 C. 22 D. 24

37、某个类中定义了成员变量

char c1, c2, c3;

byte b1, b2;

int i;

这些成员变量占用的内存空间是多少? _____。

- A. 12 B. 10 C. 14 D. 8

38、将长整型(long)数据转换为较短的整数类型(int)数据，要进行_____。

- A. 类型的自动转换 B. 类型的强制转换
C. 无需转换 D. 无法实现

39、在方法内部使用，代表对当前对象自身引用的关键字是_____。

- A. super B. This C. Super D. This

40、下列标识符定义中，不合法的是_____。

- A. int \$x; B. int _123; C. int Sum_2; D. int #dim;

41、下列选项中为单精度数的是_____。

- A. 2 B. 5.2 C. 0.2f D. 023

- 42、下列选项中属于字符串常量的是 _____。
- A. 'abc' B. "abc" C. [abc] D. (abc)
- 43、设数组 age 的定义为: int age=new int[9]; 则数组的最后一个元素的正确引用方法为_____。
- A. age[9] B. age[0] C. age[0] D. age[8]
- 44、Java 的字符类型采用的是 Unicode 编码方案,每个 Unicode 码占用_____个比特位。
- A. 8 B. 16 C. 32 D. 64
- 45、Java 中所有类的父类是_____。
- A. Father B. Lang C. Exception D. Object
- 46、类变量必须带有的修饰符是_____。
- A. static B. final C. public D. volatile
- 47、不允许作为类及类成员的访问控制符的是_____。
- A. public B. private C. static D. protected
- 48、下列_____符号修饰成员后,可以在 Java 中用类名前缀直接调用。
- A. static B. abstract C. final D. define
- 49、下列修饰符中与访问控制无关的是 _____。
- A. private B. Public C. Protected D. Final
- 50、在使用 interface 声明一个接口时,只可以使用_____修饰符修饰该接口。
- A. private B. protected C. default D. public
- 51、若需要定义一个类方法,应使用哪种修饰符? _____。
- A. static B. package C. private D. public
- 52、用来导入已定义好的类或包的语句是 _____。
- A. public B. class C. public class D. import
- 53、定义一个类,必须使用的关键字是_____。
- A. public B. class C. interface D. static
- 54、定义一个接口必须使用的关键字是_____。
- A. public B. class C. interface D. static
- 55、如下哪个是 Java 中有效的关键字_____。
- A. name B. hello C. false D. good
- 56、如下哪个是 Java 中的标识符_____。
- A. public B. super C. 3number D. width
- 57、如下哪个是 Java 中的标识符_____。
- A. fieldname B. super C. 3number D. #number
- 58、若定义有变量 float x1, x2 = 1.0F, 则下列说法正确的是_____。
- A. 变量 x1, x2 均被初始化为 1.0
- B. 变量 x1 没有被初始化, x2 被初始化为 1.0
- C. 变量 x1, x2 均未被初始化
- D. 变量 x2 没有被初始化, x1 被初始化为 1.0
- 59、关于选择结构下列哪个说法正确? _____。
- A. if 语句和 else 语句必须成对出现
- B. if 语句可以没有 else 语句对应
- C. switch 结构中每个 case 语句中必须用 break 语句
- D. switch 结构中必须有 default 语句

- 60、while 循环和 do...while 循环的区别是 _____。
- A. 没有区别，这两个结构任何情况下效果一样
 - B. while 循环比 do...while 循环执行效率高
 - C. while 循环是先循环后判断，所以循环体至少被执行一次
 - D. do...while 循环是先循环后判断，所以循环体至少被执行一次
- 61、在 switch(expression)语句中，expression 的数据型不能是 _____。
- A. char
 - B. short
 - C. double
 - D. byte
- 62、Java 语言提供的合法的关键字是 _____。
- A. Char
 - B. max
 - C. switch
 - D. Default
- 63、关于对象成员占用内存的说法哪个正确？ _____。
- A. 同一个类的对象共用同一段内存
 - B. 同一个类的对象使用不同的内存段，但静态成员共享相同的内存空间
 - C. 对象的方法不占用内存
 - D. 以上都不对
- 64、下列选项中，与成员变量共同构成一个类的是_____。
- A. 关键字
 - B. 方法
 - C. 运算符
 - D. 表达式
- 65、下列说法哪个正确？ _____。
- A. 不需要定义类，就能创建对象
 - B. 对象中必须有属性和方法
 - C. 属性可以是简单变量，也可以是一个对象
 - D. 属性必须是简单变量
- 66、在 Java 中，一个类 class 可能会有多个且多种变量，以下哪一种变量必须声明为 static 类型_____。
- A. 类变量
 - B. 成员变量
 - C. 局部变量
 - D. 所有的变量
- 67、构造函数何时被调用？ _____。
- A. 创建对象时
 - B. 类定义时
 - C. 使用对象的方法时
 - D. 使用对象的属性时
- 68、抽象方法 _____。
- A. 可以有方法体
 - B. 可以出现在非抽象类中
 - C. 是没有方法体的方法
 - D. 抽象类中的方法都是抽象方法
- 69、_____是不能被当前类的子类重新定义的方法。
- A. 抽象方法
 - B. 私有方法
 - C. 最终方法
 - D. 构造方法
- 70、下面关于类的继承性的描述中，错误的是_____。
- A. 继承是在已有的基础上生成新类的一种方法
 - B. Java 语言要求一个子类只有一个父类
 - C. 父类中成员的访问权限在子类中将被改变
 - D. 子类继承父类的所有成员，但不包括私有的成员方法
- 71、关于继承的说法正确的是 _____。
- A. 子类将继承父类所有的属性和方法。
 - B. 子类将继承父类的非私有属性和方法。
 - C. 子类只继承父类 public 方法和属性
 - D. 子类只继承父类的方法，而不继承属性

- 72、关于构造函数的说法哪个正确？_____。
- A. 一个类只能有一个构造函数
 - B. 一个类可以有多个不同名的构造函数
 - C. 构造函数与类同名
 - D. 构造函数必须自己定义，不能使用父类的构造函数
- 73、下列关于继承的描述中，正确的是_____。
- A. 在 Java 中允许多重继承
 - B. Java 的单一继承使代码更可靠
 - C. 在 Java 中一个类只能实现一个接口
 - D. 在 Java 中一个类不能同时继承一个类和实现一个接口
- 74、在 Java 中，子类可继承父类中的方法，有时子类并不想原封不动地继承父类的方法，而是想作一定的修改，这种面向对象程序特性称为_____。
- A. 隐藏
 - B. 重载
 - C. 覆盖
 - D. Java 不支持此特性
- 75、在 Java 中，一个类可同时定义许多同名的方法，这些方法的形式参数的个数、类型或顺序各不相同，传回的值也可以不相同。这种面向对象程序特性称为_____。
- A. 隐藏
 - B. 覆盖
 - C. 重载
 - D. Java 不支持此特性
- 76、覆盖与重载的关系是_____。
- A. 覆盖只有发生在父类与子类之间，而重载可以发生在同一个类中
 - B. 覆盖方法可以不同名，而重载方法必须同名
 - C. final 修饰的方法可以被覆盖，但不能被重载
 - D. 覆盖与重载是同一回事
- 77、Java 中子类可以重写（Override）其父类的同名方法，下面不符合方法重写规则的_____。
- A. 重写后的方法与被重写的方法不能具有不同的参数表
 - B. 重写后的方法不能比被重写的方法具有更严格的访问权限
 - C. 重写后的方法不能比被重写的方法产生更少的例外
 - D. 重写后的方法与被重写的方法不能具有不同类型的返回值
- 78、Java 中子类可以重载（Overload）其父类的同名方法，下面不符合方法重载规则的_____。
- A. 重载后的方法与被重载的方法不一定具有不同的参数表
 - B. 重载后的方法不一定比被重载的方法具有更严格的访问权限
 - C. 重载后的方法不一定比被重载的方法产生更多的例外
 - D. 重载后的方法与被重载的方法不一定具有不同类型的返回值
- 79、非抽象类 A 在继承抽象类 B 时，_____。
- A. 必须重写（Override）B 中所有抽象方法
 - B. 可选择重写（Override）A 需要调用的抽象方法
 - C. 必须重写（Override）B 中所有方法
 - D. 可选择重写（Override）A 需要调用的方法
- 80、下面哪个方法不是 public void mtd(){...} 的重载（Overload）方法_____。
- A. public void mtd(int m){...}

- B. **public int mtd(){...}**
C. public void mtd(float f){...}
D. public int mtd (int m, float f){...}
- 81、欲构造 ArrayList 类的一个实例，此类继承了 List 接口，下列哪个方法是正确的_____。
- A. ArrayList myList=new Object();
B. **List myList=new ArrayList();**
C. ArrayList myList=new List();
D. List myList=new List();
- 82、下列选项中关于 Java 面向对象的几个关键字的说法中，错误的是_____。
- A. implements 关键字可以一次性实现多个接口
B. this 关键字指向自己类的引用，super 关键字指向当前对象父类的引用
C. **final 关键字修饰的类无法被继承，但 final 关键字修饰的方法可以被重写**
D. 在接口中声明的方法可以不需要 abstract 关键字
- 83、关于 super 的说法正确的是_____。
- A. 是指当前对象的内存地址
B. 是指当前对象的父类对象的内存地址
C. **是指当前对象的父类**
D. 可以用在 main()方法中
- 84、关于接口哪个正确？_____。
- A. **实现一个接口必须实现接口的所有方法**
B. 一个类只能实现一个接口
C. 接口间不能有继承关系
D. 接口和抽象类是同一回事
- 85、下面关于接口的描述中，错误的是_____。
- A. **一个类只允许继承一个接口**
B. 定义接口使用的关键字是 interface
C. 在继承接口的类中通常要给出接口中定义的抽象方法的具体实现
D. 接口实际上是由常量和抽象方法构成的特殊类
- 86、下列关于抽象类的描述中，错误的是_____。
- A. 抽象类是用修饰符 abstract 说明的
B. **抽象类是不可以定义对象的**
C. 抽象类是不可以有构造方法的
D. 抽象类通常要有它的子类
- 87、下列关于构造方法的叙述中，错误的是_____。
- A. 构造方法名与类名必须相同
B. 构造方法没有返回值，且不用 void 声明
C. 构造方法只能通过 new 自动调用
D. **构造方法不可以重载，但可以继承**
- 88、对于构造函数，下列叙述不正确的是_____。
- A. 构造函数也允许重载。
B. 子类无条件地继承父类的无参构造函数。
C. **子类不允许调用父类的构造函数。**
D. 在同一个类中定义的重载构造函数可以相互调用。

- 89、若类 A 中有一个成员变量为 m，类 B 为 A 同一个包的子类，类 C 为 A 的不同包的子类，则变量 m 定义为()属性，在 B 中能访问到 m，但是在 C 中却访问不到_____。
- A. private B. public C. default D. protected
- 90、若类 A 中有一个成员变量为 m，类 B 为 A 同一个包的非子类，类 C 为 A 的不同包的子类，则变量 m 定义为()属性，在 C 中能访问到 m，但是在 B 中却访问不到_____。
- A. private B. public C. default D. protected
- 91、对于 Class T 中的成员 int i，如果 i 该类所有对象的共同属性（即所有对象该属性应相同），应当使用_____修饰。
- A. static B. public C. private D. final
- 92、下面哪些选项是正确的 main 方法说明？_____。
- A. void main() B. private static void main(String args[])
C. public main(String args[]) D. public static void main(String args[])
- 93、下列哪些方法与方法 public void sub(int a){} 为不合理的重载方法？_____。
- A. public void sub (char a) B. public int sub (int a)
C. public void sub (int a,int b) D. public void sub (float a)
- 94、为 AB 类的一个无形式参数无返回值的方法 method 书写方法头，使得使用类名 AB 作为前缀就可以调用它，该方法头的形式为 _____。
- A. static void method() B. public void method()
C. final void method() D. abstract void method()
- 95、Java 程序中以下定义正确的是_____。
- A. public class X extends Y, implements Z
B. public class X extends Y , Z
C. public class C extends X implements Y, Z
D. public implements X
- 96、若 PB 包中的类 CB，可以创建 PA 包中的类 CA 的实例，但无法访问 CA 中的方法 FA()，则 CA 和 FA() 的访问修饰符可能分别是_____。
- A. public public B. public default
C. default public D. default default
- 97、假设需要对某个字符串使用冒泡排序法对字符进行排序，不建议选用以下哪个类保存字符串_____。
- A. StringBuilder B. String C. StringBuffer D. 其他选项都可以
- 98、Java 中的抽象类 Reader 和 Writer 所处理的流是_____。
- A. 图像流 B. 对象流 C. 字节流 D. 字符流
- 99、在 Java 中，存放字符串常量的对象属于_____类对象。
- A. Character B. String C. StringBuffer D. Vector
- 100、若试图在 PB 包中实例化 PA 包中的 public 类 CA，则以下代码错误的是 Java _____。
- A. PA.CA a = new PA.CA(); B. import PA; CA a = new CA();
C. import PA.CA; CA a = new CA(); D. import PA.*; CA a = new CA();
- 101、对于接口 interface A 和 interface B 和类 Class C，以下错误的操作是_____。

- A. class C implments A; B. **class C extends A;**
C. interface B extends A; D. class C implments A,B;

102、下列叙述哪些是错误的_____。

- A. final 类不可以有子类
B.构造方法是类的一种特殊方法，其方法名必须与类名相同
C.**抽象类可以用 new 运算符创建对象**
D.内存回收程序不允许程序员直接释放内存

103、下列叙述哪些是错误的_____。

- A. abstract 类不可以用 new 和构造函数定义对象
B. **构造方法的返回值类型只能是 void 型**
C. 内存回收程序负责释放无用内存
D. Java 类只能是单继承的

104、下列叙述哪些是错误的_____。

- A. **abstract 类中只可以有 abstract 方法**
B. 构造方法的主要作用是对类的对象进行初始化工作
C. 抽象类的非抽象子类必须重写父类中的 abstract 方法
D. 匿名类可以是内部类

105、下列叙述哪些是错误的_____。

- A. public 类和没有修饰符的类有一定区别
B. Java 中类的构造方法名称必须与类的名称相同
C. 抽象类中可以有非 abstract 方法
D. **内存回收程序可以由程序员释放指定的内存对象**

106、下列叙述哪些是错误的_____。

- A. 内部类可以直接访问外部类的成员
B. 在创建对象时，系统会自动调用构造方法
C. 抽象类中可以有 abstract 方法
D. **程序员必须创建一个线程来释放内存**

107、下面的程序段的功能是_____。

```
File file1=new File("d:\\xxx\\yyy\\zzz");  
file1.mkdirs();
```

- A. 在当前目录下生成子目录: \xxx\yyy\zzz
C. 生成目录: e:\xxx\yyy\zzz
B. 在当前目录下生成文件 xxx.yyy.zzz
D. **以上说法都不对**

108、异常包含下列哪些内容? _____。

- A. **程序执行过程中遇到的事先没有预料到的情况**
B. 程序中的语法错误
C. 程序的编译错误
D. 以上都是

109、对于已经被定义过可能抛出异常的语句，在编程时_____。

- A. **必须使用 try/catch 语句处理异常，或用 throws 将其抛出**
B. 如果程序错误，必须使用 try/catch 语句处理异常

- C. 可以置之不理
D. 只能使用 try/catch 语句处理
- 110、当某一线程正处于休眠状态，而另一个线程用 Thread 类中的 interrupt() 方法中断它时，抛出的异常类型是_____。
- A. IOException B. RuntimeException
C. InterruptedException D. ClassNotFoundException
- 111、在编写异常处理的 Java 程序中，每个 catch 语句块都应该与_____语句块对应，使得用该语句块来启动 Java 的异常处理机制。
- A. if-else B. switch **C. try** D. throw
- 112、在处理异常是时，相关代码要求_____。
- A. 必须同时包含 try、catch、finally
B. **可以只包含 try、 finally，没有 catch**
C. 必须同时包含 try、catch
D. 可以只包含 try，没有 catch、finally
- 113、Java 中打开磁盘文件 c:\xt,不会用到的类是_____。
- A. InputStream B. File
C. FileReader D. **Socket**
- 114、Java 中处理中文输入，不需要用到的类是_____。
- A. BufferedReader **B. OutputStream**
C. InputStream D. System.in
- 115、下列关于 Java 语言的一些描述，错误的是_____。
- A. Java 能够自动分配与回收内存空间
B. **Java 类之间、接口之间仅支持单继承**
C. Java 丢弃了指针与操作符重载
D. Java 提供多线程之间的同步机制
- 116、假设一个无序数组 a 中，则称数组中这两个数字为逆序对，请问下列哪个排序方法的思想对于快速找到这样的逆序对的个数最有帮助_____。
- A. 堆排序 B. 冒泡排序 **C. 归并排序** D. 选择排序
- 117、如果容器组件 p 的布局是 BorderLayout，则在 p 的下边中添加一个按钮 b，应该使用的语句是_____。
- A. p.add(b); B. p.add(b,"North");
C. p.add(b,"South"); D. b.add(p,"North");
- 118、声明并创建一个按钮对象 b，应该使用的语句是_____。
- A. Button b=new Button();** B. button b=new button();
C. Button b=new b(); D. b.setLabel("确定");
- 119、Frame 对象默认的布局管理器是_____。
- A. FlowLayout **B. BorderLayout**
C. CardLayout D. null
- 120、在 Java Applet 程序用户自定义的 Applet 子类中，常常重载_____方法在 Applet 的界面中显示文字、图形和其它界面元素。
- A. start() B. stop() C. init() **D. paint()**
- 121、在 Applet 的 init()方法被调用后，接下来最先被调用的方法是_____。

- A. run() **B. start()** C. stop() D. destroy()
- 122、在 Java 中，开发图形用户界面的程序需要使用一个系统提供的类库，这个类库就是_____包。
- A. java.io **B. java.awt** C. java.applet D. java.awt.event
- 123、在 Java 图形用户界面编程中，若显示一些不需要修改的文本信息，一般是使用_____类的对象来实现。
- A. JLabel B. JButton **C. JTextArea** D. JTextField
- 124、在调用方法时，若要使方法改变实参的值，可以_____。
- A. 用基本数据类型作为参数 B. 用指针作为参数
C. 用对象作为参数 D. 以上都不可以
- 125、下列数中为八进制的是_____。
- A. 27 B. 0x25 **C. 026** D. 028
- 126、容器被重新设置大小后，B 布局管理器的容器中的组件大小不随容器大小的变化而改变？
- A. CardLayout **B. FlowLayout** C. BorderLayout D. GridLayout
- 127、下列包中，包含 Scanner 类的是_____。
- A. javax.swing B. java.lang **C. java.util** D. Java.applet
- 128、用于在子类中调用被重写父类方法的关键字是_____。
- A. this **B. super** C. This D. Super
- 129、下列不属于表达式语句的是_____。
- A. ++i; B. --j; **C. b#a;** D. b*=a;
- 130、下列为窗口事件的是_____。
- A. MouseEvent **B. WindowEvent**
C. ActionEvent D. KeyEvent
- 131、用鼠标点击菜单项 (MenuItem) 产生的事件是_____。
- A. MenuEvent **B. ActionEvent** C. KeyEvent D. MouseEvent
- 132、当一个 Applet 所在的 Web 页面被其他页面覆盖后，不可能被调用的 Applet 方法是_____。
- A. destroy() **B. init()** C. stop() D. start()
- 133、用于设置组件大小的方法是_____。
- A. paint() **B. setSize()** C. getSize() D. repaint()
- 134、点击窗口内的按钮时，产生的事件是_____。
- A. MouseEvent B. WindowEvent **C. ActionEvent** D. KeyEvent
- 135、AWT 中用来表示对话框的类是_____。
- A. Font B. Color C. Panel **D. Dialog**
- 136、下列运算中结果为 1 的是_____。
- A. 8>>1 **B. 4>>>2** C. 8<<1 D. 4<<<2
- 137、在关闭浏览器时调用，能够彻底终止 Applet 并释放该 Applet 所有资源的方法是_____。
- A. stop() **B. destroy()** C. paint() D. start()
- 138、自定义异常类的父类可以是_____。
- A. Error B. VirtualMachineError
C. Exception D. Thread
- 139、Applet 的默认布局管理器是_____。
- A. BorderLayout **B. FlowLayout**
C. GridLayout D. PanelLayout
- 140、Java 中类 ObjectOutputStream 支持对象的写操作，这是一种字节流，它的直接父类是_____。
- A. Writer B. DataOutput

C. OutputStream

D. ObjectOutputStream

141、下列_____事件表明在一个 java.awt.Component 构件之上有一个按钮按下。

A. KeyEvent

B. KeyDownEvent

C. KeyPressEvent

D. KeyTypedEvent

142、_____创建一个监听类，当鼠标移动时，它可以接受事件。

A. 通过继承 MouseListener 接口

B. 通过实现 MouseListener 接口

C. 通过继承 MouseMotionListener 接口

D. 通过实现 MouseMotionListener 接口

143、以下说法正确的是_____。

A. 一个 GridLayout 布局管理器可以将多个构件定位在多行或多列上

B. 一个 BorderLayout 布局管理器的 North 位置是放置一个 Frame 的菜单条的恰当位置

C. 一个 GridLayout 布局管理器的各单元的大小由各自所放置的构件决定

D. 一个 BorderLayout 布局管理器中的构件的大小不随容器大小的改变而改变

145、假定有一个动作事件 (ActionEvent)，_____方法可以判别出产生事件的构件。

A. public Class getClasses()

B. public Object getSource()

C. public EventObject getSource()

D. public Component getTarget()

146、_____布局管理器使用的是组件的最佳尺寸。

A. FlowLayout

B. BorderLayout

C. GridLayout

D. CardLayout

147、以下说法正确的是_____。

A. Error 类是一个 RuntimeException 异常

B. 任何抛出一个 RuntimeException 异常的语句必须包含在 try 块之内

C. 任何抛出一个 Error 对象的语句必须包含在 try 块之内

D. 任何抛出一个 Exception 异常的语句必须包含在 try 块之内

148、执行下列程序段后，b，x，y 的值正确的是_____。

```
public class Demo{
    public static void main(String[] args) {
        int x=6, y=8;
        boolean b;
        b= x < y || ++ x == --y;
        System.out.println(b );
        System.out.println(x);
        System.out.println(y);
    }
}
```

A. true , 6, 8

B. false , 7, 7

C. true, 7, 7

D. false, 6 , 8

149、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
String s = "xxxxxxxxxxxxxx#123#456#zzzzz";
int n = s.indexOf("#");
int k = s.indexOf("#", n+1);
String s2 = s.substring(n+1, k);
System.out.println(s2);
```

A. 123456

B. 123

C. xxxxxxxxxxxxxxxx

D. zzzzzz

150、关于下面的程序 Test.java 说法正确的是_____。

```
public class Test {  
    String x="1";  
    int y;  
    public static void main(String args[]) {  
        int z=2;  
        System.out.println(x+y+z);  
    }  
}
```

- A. 3 B. 102 C. 12 D. 程序有编译错误

151、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
int b[][]={{1, 1, 1}, {2,2}, {3}};  
int sum=0;  
for(int i=0; i<b.length; i++) {  
    for(int j=0; j<b[i].length; j++) {  
        sum+=b[i][j];  
    }  
}  
System.out.println("sum="+sum);
```

- A. 10 B. 6 C. 9 D. 13

152、应用程序的 main 方法中有以下语句，则执行后输出的结果是_____。

```
int[] x = {125,21,5,168,98};  
int min = x[0];  
for (int i=1; i<x.length; i++){  
    if(x[i] < min)  
        min = x[i];  
}  
System.out.println(min);
```

- A. 125 B. 5 C. 98 D. 168

153、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
Hashtable hashtable=new Hashtable();  
hashtable.put("100","aaa");  
hashtable.put("200","bbb");  
hashtable.put("300","ccc");  
System.out.println(hashtable.get("300").toString()  
    + hashtable.get("200").toString()  
    + hashtable.get("100").toString());
```

- A. aaa B. bbb C. ccc D. cccbbbaaa

154、下列程序段执行后的结果是_____。

```
String s = new String("abcdefg");  
for (int i=0; i<s.length(); i+=2){  
    System.out.print(s.charAt(i));  
}
```

}

A. aceg

B. ACEG

C. abcdefg

D. abcd

155、有整型数组：int[] x={12,35,8,7,2}；则调用方法 Arrays.sort(x)后，数组 x 中的元素值依次是_____。

A. 2 7 8 12 35

B. 12 35 8 7 2

C. 35 12 8 7 2

D. 8 7 12 35 2

156、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
String s1=new String("abc");
String s2=new String("abc");
boolean b1=s1.equals(s2);
boolean b2=(s1==s2);
System.out.print(b1+" "+b2);
```

A. true false

B. false true

C. true true

D. false false

157、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
Hashtable hashtable=new Hashtable();
hashtable.put("x","12345");
hashtable.put("y","67890");
hashtable.put("a","abcde");
System.out.println(hashtable.get("a"));
```

A. a

B. abcde

C. 12345

D. 67890

158、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
String s="12345#aaa#bbb#67890";
int n=s.indexOf("#");
int k=s.indexOf("#",n+1);
int m=s.indexOf("#",k+1);
String s2=s.substring(m+1);
System.out.println(s2);
```

A. 123456

B. 67890

C. aaa

D. bbb

159、下面程序段执行后 b 的值是_____。

```
Integer integ =new Integer(9);
boolean b = integ instanceof Object;
```

A. 9

B. true

C. 1

D. false

160、应用程序 Test.java 的源程序如下，在命令行键入：java Test aaa bb c 回车后输出的结果是_____。

```
public class Test {
    public static void main(String args[]) {
        int k1=args.length;
        int k2=args[1].length();
        System.out.print(k1+" "+k2);
    }
}
```



```
}
```

A. 3 2

B. 1 2

C. 1 3

D. 3 3

161、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
int b[][]={{1}, {2,2}, {2,2,2}};
int sum=0;
for(int i=0;i<b.length;i++) {
    for(int j=0;j<b[i].length;j++) {
        sum*=b[i][j];
    }
}
System.out.println("sum="+sum);
```

A. 32

B. 11

C. 2

D. 3

162、应用程序的 main 方法中有以下语句，则执行后输出的结果是_____。

```
int[] x={125,21,5,168,98};
int max=x[0];
for(int i=1;i<x.length;i++){
    if(x[i]> max)
        max =x[i];
}
System.out.println(max);
```

A. 125

B. 5

C. 98

D. 168

163、程序 Test.java 编译运行后输出的结果是_____。

```
public class Test {
    String s1="java";
    public static void main(String args[]) {
        int z=2;
        Test t=new Test();
        System.out.println(t.s1+z);
    }
}
```

A. java2

B. 2

C. 没有输出结果

D. java

164、应用程序的 main 方法中有以下语句，则输出的结果是_____。

```
String s1="0.5", s2="12";
double x=Double.parseDouble(s1);
int y=Integer.parseInt(s2);
System.out.println(x+y);
```

A. 12.5

B. 120.5

C. 12

D. "12.5"

165、已知定义：String s="story"; 下面哪个语句不是合法的_____。

A. s += "books";

B. s = s + 100;

C. int len = s.length;

D. String t = s + "abc";

167、下面的代码段执行之后 count 的值是什么_____。

```
int count = 1;
for (int i = 1; i <= 5; i++) {
    count += i;
}
System.out.println(count);
```

- A. 5 B. 1 C. 15 D. 16

168、假设 str1 与 str2 为两个字符串，请问下列函数实现了什么样的功能_____。

```
public int fun(String str1, String str2) {
    int[][] dp = new int[str1.length() + 1][str2.length() + 1];
    int max = 0;
    int pos = 0;
    for (int i = 1; i <= str1.length(); i++){
        for (int j = 1; j <= str2.length(); j++){
            if (str1.charAt(i - 1) == str2.charAt(j - 1))
                dp[i][j] = dp[i - 1][j - 1] + 1;
            else
                dp[i][j] = 0;
            if (dp[i][j] > max){
                max = dp[i][j];
                pos = i - 1;
            }
        }
    }
    return max;
}
```

- A. 查看 str1 中出现 str2 的位置
B. 找到 str1 需要修改多少次变成 str2
C. 找到 str1 与 str2 的最长公共子串长度
D. 找到 str1 与 str2 中最长回文子串的长度

169、下列 Java 程序实现了统计一行字符串中各类字母出现的次数，假设 s 是 java String 类型变量，则空白横线处应该补充的代码是_____。

```
_____
int letter = 0, space = 0, num = 0, other = 0;
for (int i = 0; i < chs.length; i++){
    if (chs[i] >= 65 && chs[i] <= 90 || chs[i] >= 97 && chs[i] <= 122){
        letter++;
    } else if (chs[i] == 32){
        space++;
    }
}
```

- A. Character[] chs = s.toCharArray()
B. char[] chs = s.toCharArray()
C. Character[] chs = s.toCharacter()
D. char[] chs = s.toChar()

二、填空题

- 1、开发与运行 Java 程序需要经过的三个主要步骤为 编辑源程序、编译生成字节码 和 解释运行字节码。
- 2、变量类型分 基本数据类型 和 引用数据类型。
- 3、java 中的继承是 单继承。
- 4、静态方法必须使用 static 关键字来修饰。
- 5、设 $x = 2$ ，则表达式 $(x++) / 3$ 的值是 0。
- 6、若 $x = 5$ ， $y = 10$ ，则 $x < y$ 和 $x \geq y$ 的逻辑值分别为 true 和 false。
- 7、抽象(abstract) 方法是一种仅有方法头，没有具体方法体和操作实现的方法，该方法须在抽象类之中定义。最终(final) 方法是不能被当前类的子类重新定义的方法。
- 8、创建一个名为 MyPackage 的包的语句是 package mypackage，该语句应该放在程序的位置为：应放在程序第一句。
- 9、设有数组定义：int MyIntArray[] = {10, 20, 30, 40, 50, 60, 70}; 则执行以下几个语句后的输出结果是 120。

```
int s = 0;
for (int i = 0; i < MyIntArray.length; i++)
    if (i%2 == 1)
        s += MyIntArray[i];
System.out.println(s);
```

- 9、在 Java 程序中，通过类的定义只能实现 单 重继承，但通过接口的定义可以实现 多 重继承关系。
- 10、如果将类 MyClass 声明为 public，它的文件名称必须是 MyClass.java 才能正常编译。
- 11、Java 程序中的单行注释符是 //，多行注释符是 /*...*/。
- 12、Java 中布尔类型的常量有两种，它们是 true 和 false。
- 13、Java 中用于定义小数的关键字有两个：float 和 double，后者精度高于前者。
- 14、Java 中用于两个数相等比较的运算符是：==，用于不相等比较的运算符是 <>。
- 15、在 Java 中定义一个字符串类型的变量 str 的语句是：String str，定义一个具有 10 个元素的整型数组 a 的语句是：int [] r=new int [10];。
- 16、导入 mypackage 包中的所类的命令是 import mypackage.*;。
- 17、当声明一个数组 int arr[] = new int[5]; 时，这代表这个数组所保存的变量类型是 int，数组名是 arr，数组的大小为 5，数组元素下标的使用范围是 0~4。
- 18、假设 $x=13$ ， $y=4$ ，则表达式 $x\%y \neq 0$ 的值是 true，其数据类型是 boolean。
- 19、异常处理是由 try、catch 和 finally 块三个关键所组成的程序块。
- 20、以下程序段的输出结果是 三角形。

```
int x = 5, y = 6, z = 4;
if (x + y > z && x + z > y && z + y > x)
    System.out.println("三角形");
else
    System.out.println("不是三角形");
```

- 21、下面程序段的执行结果是 6 5 4 3 2

```
int a[] = { 2, 3, 4, 5, 6 };
```

```
for (int i = a.length-1; i >= 0; i--)
```

```
System.out.print(a[i] + "");
```

- 22、面向对象的三大特性，是封装性、继承性、多态性。
- 23、被关键字final修饰的方法是不能被当前类的子类重新定义的方法。
- 24、可以通过import关键字引用其它包中定义的类和接口。
- 25、Java 中，引用父类中的成员需要使用关键字super。
- 26、针对类、成员方法和属性，Java 提供了 4 种访问控制权限，分别是private、public、protected 和 default。
- 27、在 Java 中解决成员变量与局部变量名称冲突时，可以使用this关键字。
- 28、一个类中可以定义多个构造方法，只要每个构造方法的参数类型或参数个数不同，即可实现重载。

注意：选择和填空相互可以转换，由于篇幅就不赘述！

三、判断题

1. Java 程序里,创建新的类对象用关键字 new,回收无用的类对象使用关键字 free。(F)
2. 有的类定义时可以不定义构造函数,所以构造函数不是必需的。(F)
3. 类及其属性、方法可以同时有一个以上的修饰符来修饰。(T)
4. Java 的屏幕坐标是以像素为单位,容器的左下角被定为坐标的起点。(F)
5. 抽象方法必须在抽象类中,所以抽象类中的方法都必须是抽象方法。(F)
6. Final 类中的属性和方法都必须被 final 修饰符修饰。(F)
7. 最终类不能派生子类,最终方法不能被覆盖。(T)
8. 子类要调用父类的方法,必须使用 super 关键字。(T)
9. 一个 Java 类可以有多个父类。(F)
10. 如果 p 是父类 Parent 的对象,而 c 是子类 Child 的对象,则语句 c=p 是正确的。(F)
11. 一个类如果实现了某个接口,那么它必须重载该接口中的所有方法。(T)
12. 当一个方法在运行过程中产生一个异常,则这个方法会终止,但是整个程序不一定终止运行。(T)
13. 接口是特殊的类,所以接口也可以继承,子接口将继承父接口的所有常量和抽象方法。(T)
14. 用 “+” 可以实现字符串的拼接,用 “-” 可以从一个字符串中去除一个字符子串。(F)
15. 一个容器中可以混合使用多种布局策略。(T)
16. 在面向对象方法中,类的实例称为对象。(T)
17. Java 的源代码中定义几个类,编译结果就生成几个以.class 为后缀的字节码文件。(T)
18. Java Applet 是由独立的解释器程序来运行的。(F)
19. Java Applet 只能在图形界面下工作。(T)
20. Java 的字符类型采用的是 ASCII 编码。(F)
21. Java 的各种数据类型占用固定长度,与具体的软硬件平台环境无关。(T)
22. 容器是用来组织其他界面成分和元素的单元,它不能嵌套其他容器。(F)
23. Java 是不区分大小写的语言。(F)
24. Java 的源代码中定义几个类,编译结果就生成几个以.class 为后缀的字节码文件。(T)
25. Java Application 是由独立的解释器程序来运行的。(F)
16. Java 语言是一种完全的面向对象的程序设计语言。(T)
27. Java Application 只能在字符界面下工作。(F)
28. Java 的字符类型采用的是 Unicode 编码,每个 Unicode 码占 16 个比特。(T)
29. Java 源程序是由类定义组成的,每个程序可以定义若干个类,但只有一个类是主类。(T)
30. 子类的域和方法的数目一定大于等于父类的域和方法的数目。(F)
31. 类在实现接口的方法时,必须显示地使用 public 修饰符。(T)
32. 一个类只能有一个父类,但一个接口可以有一个以上的父接口。(T)
33. 程序中抛出异常时(throw ...),只能抛出自己定义的异常对象。(F)
34. System 类中的 in 是一个输入对象。(T)
35. 实现一个接口,则在类中一定要实现接口中的所有方法。(T)
36. 所有的文件输入/输出流都继承于 InputStream 类/OutputStream 类。(F)
37. 一个异常处理中 finally 语句块只能有一个或者可以没有。(F)
38. 语句 import java.applet.Applet; 最后的 Applet 是代表类(class)。(T)
39. 类体中 private 修饰的变量在本类中能访问,类生成的对象也能访问。(F)
40. 线程使用 sleep()方法休眠后,可以用 notify()方法唤醒。(F)
41. 一个异常处理中 finally 语句块只能有一个或者可以没有。(T)
42. 用 Javac 编译 Java 源文件后得到代码叫字节码。(T)
43. 数组、类和接口都是引用数据类型。(F)
44. 一个 Applet 编译后的类名是 Test.class,运行此小程序的命令是 Java Test。(F)
45. Java 语言中一个 Char 类型的数据占用 2 个字节大小的内存空间。(T)
46. 编译语句 Double aDouble=37.2D 不会出现编译错误。(F)

- 47、编译语句 `int aInt=66666` 不会出现编译错误。 (T)
- 48、判断下面说法的正误：
- ```
if(true){
 int i=17;
}
System.out.println(i);
```
- 运行此程序的结果是输出 17。 ( F )
- 49、`float x=26f; int y=26; int z=x/y;` 以上语句能正常编译和运行。 ( F )
- 50、判断下面说法的正误：
- ```
int x=9;  
if(x>8 and x<10)  
    System.out.println("true");
```
- 以上语句运行的结果是显示 true。 (T)
- 51、判断下面说法的正误：
- ```
String str;
System.out.println(str.length());
```
- 以上语句运行的结果是显示 0 。 ( F )
- 52、抽象类不能实例化。 ( T )
- 53、一个方法最多能有一个 `return` 语句。 ( F )
- 54、接口名可以作为方法返回值的类型。 ( T )
- 55、类 A 和类 B 位于同一个包中，则除了私有成员，类 A 可以访问类 B 的所有其他成员。 ( T )
- 56、Java 语言的标识符的命名是区分大小写的。 ( T )
- 57、Java 源程序的文件名与 `public` 类型的类名必须相同。 ( T )
- 58、构造函数是类的一种特殊方法，它的方法名必须与类名相同 ( T )
- 59、编译语句 `double a=37.2;` 不会出现编译错误。 ( T )
- 60、定义完数组后，我们可以根据实际需要动态调整数组大小。 ( F )
- 61、子类能继承父类的所有方法和属性。 ( F )
- 62、方法的重载是指子类 and 父类具有相同的名字、相同的参数表。 ( F )
- 63、在 Java 中子类只能继承父类 `public` 方法和属性。 ( F )



## 四、简答题

### 1、类体中的方法包含哪些分类？

答：构造方法、普通方法、静态方法。

### 2、请写出基本数据类型

答：byte, short, int, long, float, double, boolean, char

### 3、简述创建对象的过程

答：

- 1) 声明：声明一个对象，包括对象名称和对象类型。
- 2) 实例化：使用关键字 **new** 来创建一个对象。
- 3) 初始化：使用 **new** 创建对象时，会调用构造方法初始化对象。

### 4、简述 Java 中异常处理的机制？

答：

首先 Java 的异常是面向对象的。一个 Java 的 **Exception** 是一个描述异常情况的对象。当出现异常情况时，一个 **Exception** 对象就产生了，并放到异常的成员函数里。

- 1) Java 的异常处理是通过 5 个关键词来实现的：**try**, **catch**, **throw**, **throws** 和 **finally**。
- 2) 异常处理：在 Java 语言的错误处理结构由 **try**, **catch**, **finally** 三个块组成。其中 **try** 块存放将可能发生异常的 Java 语言，并管理相关的异常指针；**catch** 块紧跟在 **try** 块后面，用来激发被捕获的异常；**finally** 块包含清除程序没有释放的资源，句柄等。不管 **try** 块中的代码如何退出，都将执行 **finally** 块。
- 3) 抛出异常：Java 语言可以不在方法中直接捕获，而用 **throw** 语句将异常抛给上层的调用者。**Throw** 语句就是来明确地抛出一个异常；首先你必需得到一个 **Throwable** 的实例句柄，通过参数传到 **catch** 中，或者采用 **new** 操作符来创建一个。

### 5、Java 中，请说明类变量与实例变量的区别？

答：

- 1) 类变量也叫静态变量，也就是在变量前加了 **static** 的变量；
- 2) 实例变量也叫对象变量，即没加 **static** 的变量；
- 3) 区别在于：类变量是所有对象共有，其中一个对象将它值改变，其他对象得到的就是改变后的结果；而实例变量则属对象私有，某一个对象将其值改变，不影响其他对象

### 6、什么是类方法和实例方法？它们的区别是什么？

答：

- 1) 用修饰符 **static** 说明的方法是类方法，其它的是实例方法。当一个类创建了一个对象后，这个对象就可以调用该类的方法。
  - 2) 区别在于类方法不仅可以由对象调用而且还可以直接由类名调用，而实例方法不能由类名调用。
- 7、什么是多态？多态主要的应用是什么？
- 1) 多态就是指同一个方法，不同的实现。多态分为编译时的多态和运行时的多态，编译时的多态通过方法的重载实现。运行时的多态通过方法的重写实现。
  - 2) 多态主要运用于增加程序的灵活性，增加程序的可扩展性。

## 8、抽象类（abstract class）和接口（interface）有什么异同？

答：相同点：

- 1、都不能被实例化
- 2、接口的实现类或者抽象类的子类都必须实现了接口或抽象类中的方法后才可以被实例化

不同点：

- 1) 抽象类可以包含非抽象的方法，而接口中的方法必须是抽象的。
- 2) 继承抽象类在 Java 语言体系中体现一种继承关系，在合理的继承关系中，父类和派生类比如按存在 is-a 关系。而实现接口则体现一种 has-a 关系，实现者仅仅是实现了 interface 定义的契约而已。
- 3) 抽象类中可以有成员变量，而接口中不能有成员变量。
- 4) 抽象类和接口中都可以包含静态成员变量，抽象类中的静态成员变量的属性可以是任意的，而接口中的静态成员变量默认且必须是 public static final。
- 5) 接口中的抽象方法默认且必须都是 public 的，而抽象类中的抽象方法可以是 public、protect 或默认。
- 6) 抽象类中可以有构造方法，但是不能实例化。
- 7) 抽象类中可以有静态方法（非抽象的），而接口中不能有静态方法。
- 8) 抽象类中的抽象方法不能用 static、synchronized 和 native 连用。

## 9、Java 中，请说明 final 关键字的使用？

答：

- 1) 用来修饰数据，包括成员变量和局部变量，该变量只能被赋值一次且它的值无法被改变。  
对于成员变量来讲，我们必须在声明时或者构造方法中对它赋值；
- 2) 用来修饰方法参数，表示在变量的生存期中它的值不能被改变；
- 3) 修饰方法，表示该方法无法被重写；
- 4) 修饰类，表示该类无法被继承。

## 10、简述构造函数特点？

答：（1）构造方法的方法名必须与类名相同；

（2）构造方法没有返回值类型，可以有 return，但是不能有返回值，在这里只是表示结束，并不是返回的表示，一般也不使用 return；

（3）构造方法的主要作用是完成对象的初始化工作，它能够把定义对象时的参数传给对象的域；

（4）一个类可以定义多个构造方法，如果在定义类时没有定义构造方法，则编译系统会自动插入一个无参数的默认构造方法，这个构造方法不执行任何代码；

（5）构造方法可以重载，以参数的个数，类型，顺序。

## 11、简述继承的好处？

答：（1）代码共享，减少创建类的工作量，每个子类都拥有父类的方法和属性；

（2）提高代码的重用性；

（3）提高代码的可扩展性。

## 12、简述 Integer 与 int 的区别？

答： (1) Integer 是 int 的包装类；int 是基本数据类型；  
(2) Integer 变量必须实例化后才能使用；int 变量不需要；  
(3) Integer 实际是对象的引用，指向此 new 的 Integer 对象；int 是直接存储数据值；  
(4) Integer 的默认值是 null；int 的默认值是 0。

## 13、简述方法的重写和重载的区别？

答：

override（重写）

- (1) 方法名、参数、返回值相同；
- (2) 重写方法的访问修饰符一定要大于或等于被重写的方法；
- (3) 重写方法不能抛出新的异常或者抛出比被重写方法更宽泛的异常；
- (4) 存在于父类和子类之间；
- (5) 被 final 修饰的方法不能重写。

overload（重载）

- (1) 参数类型、个数、顺序至少有一个不相同；
- (2) 不能重载只有返回值不同的方法名；
- (3) 存在于同类中。

## 14、请简述重载和重写的区别？

答：

- 1) 方法的重写 Override 和重载 Overload 是 Java 多态性的不同表现。
- 2) 重写 Override 是父类与子类之间多态性的一种表现，重载 Overload 是一个类中多态性的一种表现。
- 3) 如果在子类中定义某方法与其父类有相同的名称和参数，我们说该方法被重写（Override）。子类的对象使用这个方法时，将调用子类中的定义，对它而言，父类中的定义如同被“屏蔽”了。
- 4) 如果在一个类中定义了多个同名的方法，它们或有不同的参数个数或有不同的参数类型，则称为方法的重载 (Overload)。Overload 的方法是可以改变返回值的类型。

## 15、请说出作用域 public, private, protected, 以及不写时的区别？

答：

- 1) 用 public 修饰的类、类属变量及方法，包内及包外的任何类（包括子类和普通类）均可以访问。
- 2) 用 private 修饰的类、类属变量及方法，只有本类可以访问，而包内包外的任何类均不能访问它。
- 3) 用 protected 修饰的类、类属变量及方法，包内的任何类及包外那些继承了该类的子类才能访问，protected 重点突出继承。
- 4) 如果一个类、类属变量及方法没有用任何修饰符（即没有用 public、protected 及 private 中任何一种修饰），则其访问权限为 default（默认访问权限）。默认访问权限的类、类属变量及方法，包内的任何类（包括继承了此类的子类）都可以访问它，而对于包外的任何类都不能访问它（包括包外继承了此类的子类）。default 重点突出包。

## 五、程序阅读题

1、阅读下面程序 Test.java，写出运行结果。

```
import java.io.*;
public class Demo{
 public static void main(String[] args){
loop:
 for(int i=2; i<10; i++) {
 for(int j=2; j<i; j++)
 if(i%j == 0)
 continue loop;
 System.out.print(i+" ");
 }
 }
}
```

运行结果： 2 3 5 7。

2、阅读下面程序 Test.java，写出运行结果。

```
import java.io.* ;
public class Test{
 public static void main(String args[]){
 int i, s=0;
 int a[] = {10, 20, 30, 40, 50, 60, 70, 80, 90};
 for(i=0; i<a.length; i++)
 if(a[i]%3==0) s+=a[i];
 System.out.println("s="+s);
 }
}
```

运行结果： S=180。

3、阅读下面程序 Test.java，写出运行结果。

```
import java.io.* ;
public class Test{
 public static void main(String args[]){
 int i, s=0;
 int a[] = {10, 20, 30, 40, 50, 60, 70, 80, 90};
 for(i=0; i<a.length; i++)
 if(i%3 == 0) s+=a[i];
 System.out.println("s="+s);
 }
}
```

运行结果： S=120。

4、阅读下面程序 Test.java，请写出该程序完成的功能。

```
import java.io.*;
public class Test{
 public static void main(String args[]){
 int i,max,min;
 int a[] = {12,67,8,98,23,56,124,55,99,100};
 max= min= a[0];
 for(i=1; i<a.length; i++){
 if(a[i]< min)
 min = a[i];
 if(a[i]> max)
 max = a[i];
 }
 }
}
```

该程序完成的功能： 在数组中查找并输出最大值和最小值。 r

5、阅读下面程序 Test.java，写出运行结果。

```
import java.io.*;
public class Test{
 public static void main(String args[]) {
 System.out.println("a="+a+"\nb="+b);
 }
}
class SubClass extends SuperClass{
 int c;
 SubClass(int aa, int bb, int cc) {
 super(aa, bb);
 c=cc;
 }
}
class SubSubClass extends SubClass{
 int a;
 SubSubClass(int aa, int bb, int cc) {
 super(aa, bb, cc);
 A = aa+bb+cc;
 }
 void show(){
 System.out.println("a="+a+"\nb="+b+"\nc="+c);
 }
}
```

运行结果： a=60。

b=20。

c=30。

6、阅读下面程序 Person.java，写出运行结果。

```
public class Person {
 String name;
 int age;
 public Person(String name, int age) {
 this.name = name;
 this.age = age;
 }
 public static void main(String[] args) {
 Person c = new Person("Peter", 17);
 System.out.println(c.name + " is " + c.age + " years old!");
 }
}
```

运行结果: Peter is 17 years old!。

7、阅读下面程序 CourseTest.java，写出运行结果。

```
public class Course {
 private String cNumber;
 private String cName;
 private int cUnit;
 public Course(String number, String name, int unit) {
 cNumber = number;
 cName = name;
 cUnit = unit;
 }
 public void printCourseInfo() {
 System.out.println("课程号:" + cNumber + " 课程名:" + cName
 + " 学分:" + cUnit);
 }
}

class CourseTest {
 public static void main(String[] args) {
 Course c;
 c = new Course("101", "ASP", 3);
 c.printCourseInfo();
 }
}
```

运行结果: 课程号: 101 课程名: ASP 学分: 3。



8、阅读下面程序 Tom.java，写出运行结果。

```
public class Tom {
 private float weight;
 private static String name;
 public void setWeight(float weight) {
 this.weight = weight;
 }
 private void out() {
 System.out.println(name + "体重: " + weight + "斤");
 }
 public static void main(String[] args) {
 Tom.name = "汤姆猫";
 Tom cat = new Tom();
 cat.setWeight(20);
 cat.out();
 }
}
```

运行结果: 汤姆猫体重: 20.0 斤。

9、阅读下面程序 Father.java，写出运行结果。

```
public class Father {
 String name, address, tel;
 int age;
 public Father(String name, int age) {
 this.name = name;
 this.age = age;
 }
 void out() {
 System.out.print("姓名:" + name);
 System.out.print(" 年龄:" + age);
 }
 void outOther() {
 System.out.print(" 家庭住址:" + address);
 System.out.print(" 电话:" + tel);
 }
}

class Son extends Father {
 String school;
 public Son(String name, int age) {
 super(name, age);
 }
}
```

```

 void out() {
 super.out();
 super.outOther();
 System.out.println(" 学校: " + school);
 }
 public static void main(String args[]) {
 Son son = new Son("Tom", 15);
 son.address = "金水区";
 son.school = "九中";
 son.tel = "66123456";
 son.out();
 }
}

```

运行结果: 姓名: Tom 年龄: 15 家庭住址: 金水区 电话: 66123456 学校: 九中。

10、阅读下面程序 MyClass.java，写出运行结果。

```

public class MyClass {
 int a[] = {1, 2, 3, 4, 5};
 void out() {
 for (int j = 0; j < a.length; j++)
 System.out.print(a[j] + "");
 }
 public static void main(String[] args) {
 MyClass my = new MyClass();
 my.out();
 }
}

```

运行结果: 1 2 3 4 5。

11、阅读下面程序 Base.java，写出运行结果。

```

public class Base{
 public static void main(String[] args) {
 int i=4;
 switch(i){
 default: System.out.print("default");
 case 0: System.out.print("Zero");
 case 1: System.out.print("One"); break;
 case 2: System.out.print("Two");
 }
 }
}

```

运行结果: defaultZeroOne。

12、阅读下面的程序代码，写出运行结果。

```
String s1 = new String("abcde");
String s2 = new String("abcde");
boolean b1= s1.equals(s2);
boolean b2 = s1== s2;
System.out.print(b1+" "+b2);
```

运行结果: True false

13、阅读下面的程序，写出运行结果。

```
import java.io.*;
public class Test {
 public static void main(String args[]) throws IOException {
 BufferedReader buf=new BufferedReader(
 new InputStreamReader(System.in));
 while(true) {
 String str = buf.readLine();
 if(str.equals("quit"))
 break;
 int x=Integer.parseInt(str);
 System.out.println(x*x);
 }
 }
}
```

从键盘输入 10，回车后输出的结果: 100。

4、阅读下面的程序，写出运行结果。

```
import java.util.*;
public class T{
 public static void main(String args[]) {
 Set set = new TreeSet();
 set.add(new Integer(10));
 set.add(new Integer(5));
 set.add(new Integer(15));
 set.add(new Integer(5));
 set.add(new Integer(10));
 System.out.println("size = " + set.size());
 Iterator it=set.iterator();
 while(it.hasNext()){
 System.out.print(it.next()+" ");
 }
 }
}
```

运行结果: Size=3 5 10 15。

5、阅读下面的程序，从键盘输入 5，回车后输出的结果\_\_\_\_\_。

```
import java.io.*;
public class Test {
 public static void main(String args[]) throws IOException {
 BufferedReader buf=new BufferedReader(
 new InputStreamReader(System.in));
 while(true) {
 String str=buf.readLine();
 if(str.equals("quit"))
 break;
 int x=Integer.parseInt(str);
 System.out.println(x*x);
 }
 }
}
```

**注意：书上类似的题目，由于篇幅就不赘述！**

## 六、程序填空题

### 1、阅读程序填空题

```
public class Sum{
 public static void main(String [] args){
 int j=10;
 System.out.println("j is : "+j);
 calculate(j);
 System.out.println("At last, j is : "+j);
 }
 static void calculate (int j){
 for (int i = 0;i<10;i++)
 j++;
 System.out.println("j in calculate() is: "+j);
 }
}
```

输出结果为:

|                       |                 |
|-----------------------|-----------------|
| j is :                | (1)_____10_____ |
| j in calculate() is : | (2)_____20_____ |
| At last j is :        | (3)_____10_____ |

### 2、按要求填空

```
abstract class SuperAbstract{
 void a(){...}
 abstract void b();
 abstract int c(int i);
}
interface AsSuper{
 void x();
}
abstract class SubAbstract extends SuperAbstract implements AsSuper
{
 public void b(){...}
 abstract String f();
}
public class InheritAbstract extends SubAbstract{
 public void x(){...}
 public int c(int i) {...}
 public String f(){...}
 public static void main(String args[]){
 InheritAbstract instance=new InheritAbstract();
 instance.x();
 }
}
```

```

 instance.a();
 instance.b();
 instance.c(100);
 System.out.println(instance.f());
 }
}

```

在以上这段程序中：

抽象类有：SuperAbstract 和 (1) (写出类名)SuperAbstract

非抽象类有： (2) (写出类名)InheritAbstract

接口有： (3) (写出接口名)AsSuper

AsSuper 中的 x()方法是 (4) 抽象 方法，所以在 InheritAbstract 中必须对它进行 (5) 覆盖和实现。

### 3、按注释完成程序

```

public class Leaf {
 private int i = 0; //此属性值用于检验
 Leaf increment(){ //定义方法 increment(), 返回值是 Leaf 类的对象
 i++;
 return (1) this; //将当前对象的地址作为返回值返回
 }
 void print() {
 System.out.println(" i = " + i);
 }
 public static void main(String args[]){
 Leaf x = (2) new Leaf(); //创建 Leaf 类的对象 x
 x.increment().increment().increment().print();
 } //多次调用方法 increment(), 返回的都是 x 的地址, i 值表示调用次数
}

```

输出结果为 i = (3) 3



## 七、程序设计题

1、定义一个类，类名为 Test1，求 1 到 10 的阶乘，请写出完整的程序代码。

答：

```
public class Test1{
 public static void main(String[] argv){
 long result =1;
 for(int i=1; i<=10; i++){
 result = result*i;
 }
 System.out.println("1 到 10 的阶乘是: " + result);
 }
}
```

2、定义一个类，类名为 Test2，实现计算“1+3+5+7+...+99”的值，请写出完整的程序代码，要求如下。

(1) 使用循环语句实现自然数 1~99 的遍历。

(2) 在遍历过程中，通过条件判断当前遍历的书是否为奇数，如果是就累加，否则不加。

```
public class Test2{
 public static void main(String[] args) {
 int sum = 0;
 for (int i = 1; i < 100; i++) {
 if (i % 2 != 0)
 sum += i;
 }
 System.out.println(sum);
 }
}
```

3、定义一个类，类名为 Test3，求 1+2+3+……100 的奇数的和，请写出完整的程序代码。

```
public class Test3{
 public static void main(String[] argv){
 Long result =1;
 for(int i =1;i<=100;i++){
 if(i % 2 == 0)
 continue;
 result = result+i;
 }
 System.out.println("1 到 100 奇数的和是: "+result);
 }
}
```

#### 4、定义一个类，类名为 Test4，找出 100 以内所有的素数（质数）？

```
public class Test4{
 public static void main(String[] argv){
 //什么是质数？ 除了1和它本身能整除的自然数。例如：2、3、5、7、11.....
 for (int i = 2; i < 100; i++) {
 //记录i的约束的个数（从2开始，到i-1为止）
 int number = 0;
 for (int j = 2; j < i; j++) {
 if (i%j==0){
 number++;
 }
 }
 if (number==0){
 System.out.println(i);
 }
 }
 }
}
```

#### 5、问题描述：

给定一个长度为 n 的数列，将这个数列按从小到大的顺序排列。1<=n<=200

##### 输入格式：

第一行为一个整数 n。

第二行包含 n 个整数，为待排序的数，每个整数的绝对值小于 10000。

##### 输出格式：

输出一行，按从小到大的顺序输出排序后的数列。

##### 样例输入：

5

8 3 6 4 9

##### 样例输出：

3 4 6 8 9

请根据上述要求，定义一个类，类名为 Test5，写出完整的程序代码。

答：

```
import java.util.Arrays;
import java.util.Scanner;
public class Test5{
 public static void main(String[] args) {
 Scanner sc=new Scanner(System.in);
 int n = sc.nextInt();
 int arr[]=new int[n];
 for(int i=0;i<n;i++){
 arr[i]=sc.nextInt();
 }
 Arrays.sort(arr);
 for(int i=0;i<n;i++){
 System.out.print(arr[i]+" ");
 }
 }
}
```

```
}
```

## 6、按以下要求编写程序

- (1) 创建一个 Rectangle 类，添加 width 和 height 两个成员变量；
- (2) 在 Rectangle 中添加两种方法分别计算矩形的周长和面积；
- (3) 编程利用 Rectangle 输出一个矩形的周长和面积。

答：

```
public class Rectangle {
 float width, height;
 public Rectangle(float width, float height) {
 this.width = width;
 this.height = height;
 }
 public float getLength(){
 return (this.width + this.height) * 2;
 }
 public float getArea(){
 return this.width * this.height;
 }
 public static void main(String [] args) {
 Rectangle rect = new Rectangle(10, 20);
 System.out.println("周长是: " + rect.getLength());
 System.out.println("面积是: " + rect.getArea());
 }
}
```

## 7、编写一个程序，要求创建一个 Student 类，添加 name 和 age 属性，为该属性自动添加相应的 getter 和 setter 方法，并给出有参和无参的构造方法。

答：

```
public class Student {
 private String name;
 private int age;

 public Student() {
 }
 public Student(String name, int age) {
 super();
 this.name = name;
 this.age = age;
 }
 public String getName() {
 return name;
 }
}
```

```

 public void setName(String name) {
 this.name = name;
 }
 public int getAge() {
 return age;
 }
 public void setAge(int age) {
 this.age = age;
 }
}

```

8、编写一个类，类中定义一个静态方法，用于求两个整数的和。请按照以下要求设计一个测试类 Demo，并进行测试。要求如下：

- (1) Demo 类中有一个静态方法 `getSum(int a, int b)` 该方法用户返回参数 a、b 两个整数的和；
- (2) 在 `main()` 方法中调用 `getSum` 方法并输出计算结果。

答：

```

public class Demo {
 public static int getSum (int a, int b) {
 return a + b;
 }
 public static void main(String[] args) {
 int result = Demo01.getSum(2, 3);
 System.out.println(result);
 }
}

```

9、编写一个程序，获取 Demo.docx 文件的扩展名。

```

public class Demo {
 public static void main(String[] args) {
 System.out.println(getExtname("Demo.docx"));
 }
 public static String getExtname(String filename){
 int index = filename.lastIndexOf(".");
 String extname = filename.substring(index+1);
 return extname;
 }
}

```

10、编写一个程序，创建两个线程，要求分别输出 26 个字母。在输出结果时，要显示是哪个线程输出的字母。

```
public class Demo01{
 public static void main(String[] args) {
 Test t1 = new Test("线程一");
 Test t2 = new Test("线程二");
 t1.start();
 t2.start();
 }
}

class Test extends Thread{
 private char ch = 'a';
 public Test(String name){
 setName(name);
 }
 public void run(){
 char temp;
 for (int i = 0; i < 26; i++) {
 temp = (char) (ch+i);
 System.out.println(this.getName() + ":" + temp);
 }
 }
}
```

11、定义一个抽象类 Car，在该类中包含一个抽象方法 run()。

(1) 分别定义一个 Bike 类和 Bus 类继承自 Car，并在重写的 run()方法中分别输出一句话；

(2) 定义测试类，调用 Bike 类和 Bus 类中的方法。

```
abstract class Car{
 abstract void run();
}

class Bike extends Car{
 void run() {
 System.out.println("自行车在行驶");
 }
}

class Bus extends Car{
 void run() {
 System.out.println("公交车在行驶");
 }
}

public class Demo {
```

```

 public static void main(String[] args) {
 Bike bike = new Bike();
 bike.run();
 Bus bus = new Bus();
 bus.run();
 }
 }
}

```

**12、按以下要求编写程序**

- (1) 编写 Animal 接口，接口中声明 run() 方法；
- (2) 定义 Bird 类和 Fish 类实现 Animal 接口；
- (3) 编写 Bird 类和 Fish 类的测试程序，并调用其中的 run()方法。

答：

```

public interface Animal {
 void run();
}

class Bird implements Animal {
 public void run() {
 System.out.println("鸟儿在飞...");
 }
}

class Fish implements Animal {
 public void run() {
 System.out.println("鱼儿在游...");
 }
}

public class TestAnimal {
 public static void main(String[] args) {
 Bird bird = new Bird();
 bird.run();
 Fish fish = new Fish();
 fish.run();
 }
}

```

13、编写一个 Frame 框架应用程序，要求如下：

- (1) 在窗口设置两个菜单“文件”、“编辑”；
- (2) 在“文件”菜单里添加三个菜单项“打开”、“保存”、“关闭”；
- (3) 在“编辑”菜单里添加两个菜单项“复制”、“粘贴”；
- (4) 点击关闭菜单项时，使程序关闭。

答：

```
import java.awt.Color;
import java.awt.Frame;
import java.awt.Menu;
import java.awt.MenuBar;
import java.awt.MenuItem;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

//主窗体
public class MyFrame extends Frame {
 private static final long serialVersionUID = 6895463895656626294L;
 private Frame frmMain; // 窗体
 MenuBar mb; // 菜单栏
 Menu mnuFile; // "文件"菜单
 Menu mnuEdit; // "编辑"菜单
 MenuItem miOpen; // "打开"菜单项
 MenuItem miSave; // "保存"菜单项
 MenuItem miClose; // "关闭"菜单项
 MenuItem miCopy; // "复制"菜单项
 MenuItem miPaste; // "粘贴"菜单项

 public MyFrame() {
 frmMain = new Frame("主窗体");

 // 创建菜单栏
 mb = new MenuBar();

 // 创建菜单
 mnuFile = new Menu("文件");
 mnuEdit = new Menu("编辑");

 // 创建菜单项
 miOpen = new MenuItem("打开");
 miSave = new MenuItem("保存");
```

```

 miClose = new MenuItem("关闭");
 miCopy = new MenuItem("复制");
 miPaste = new MenuItem("粘贴");
 }

 public void showFrame() {
 frmMain.setSize(800, 600);
 frmMain.setLocation(100, 100);
 frmMain.setBackground(Color.white);
 frmMain.setVisible(true);
 frmMain.setLayout(null);

 frmMain.addWindowListener(new WindowHandler()); // 注册窗口监听器

 // 将菜单栏放入窗体中
 frmMain.setMenuBar(mb);

 // 将菜单放入菜单栏中
 mb.add(mnuFile);
 mb.add(mnuEdit);

 // 将菜单项放入菜单中
 mnuFile.add(miOpen);
 mnuFile.add(miSave);
 mnuFile.add(miClose);

 mnuEdit.add(miCopy);
 mnuEdit.add(miPaste);

 // 注册动作事件监听器
 miClose.setActionCommand("miClose_Clicked");
 miClose.addActionListener(new MenuHandler());
 }

 // 定义一个内部类，在这个类中编写窗口关闭事件处理程序。
 private class WindowHandler extends WindowAdapter {
 public void windowClosing(WindowEvent e) {
 System.exit(0);
 // 退出应用程序，状态码 0 表示正常终止，非 0 表示异常终止。
 }
 }

 // 定义一个内部类，在这个类中编写动作事件处理程序。

```



```
private class MenuHandler implements ActionListener {
 public void actionPerformed(ActionEvent e) {
 if (e.getActionCommand() == "miClose_Clicked") {
 System.exit(0); // 退出应用程序
 }
 }
}
}

public class TestMyFrame {
 public static void main(String[] args) {
 // 启动主窗体
 MyFrame guiWindow = new MyFrame();
 guiWindow.showFrame();
 }
}
```

注意：教材上相关章节编程题自行巩固，由于篇幅就不赘述！